

Column-Generation for Design of Survivable Electricity Distribution Networks

Kavinesh Singh
Andy Philpott
Engineering Science Department
University of Auckland
Auckland, New Zealand

Kevin Wood
Operations Research Department
Naval Postgraduate School
Monterey, California, USA

Abstract

We investigate the problem of designing survivable electricity distribution networks subject to multiple, non-simultaneous link failures under a radial-network operating configuration. We formulate this problem as a two-stage stochastic mixed-integer program in which first-stage decisions expand capacity; recourse decisions configure the network to operate as a tree and to meet demand, by opening and closing electrical switches. Dantzig-Wolfe decomposition of this formulation leads to (a) a master problem comprising binary capacity-expansion and high-level operating decisions; and (b) mixed-integer, column-generating subproblems which represent deterministic capacity-expansion models. A “super-arc representation” of the network significantly reduces the number of binary variables, and provides a tighter linear-programming relaxation for the subproblems. Column generation with super-arc subproblems solves the model significantly faster than CPLEX can solve the original, extensive model.

1 Introduction

This paper focuses on the problem of designing survivable electricity distribution networks. These networks transport electricity from drop-off points of a high-voltage transmission grid to local residential, commercial, and industrial consumers. In essence, distribution networks consist of (a) one or more power sources, i.e., drop-off points where the high-voltage electricity is stepped down through transformers to a lower voltage for distribution, (b) demand points, (c) junctions, (d) switches, and (e) interconnecting power lines. An urban distribution network may contain hundreds or even thousands of such components. Such a network is “survivable” if it can recover from a “line fault,” i.e., the failure of a cable or associated equipment.

Distribution networks can be operated in several different configurations including mesh, interconnected, link arrangement, open loop, and radial (Lakervi and Holmes 1995). We consider networks, with underlying mesh structure, that operate in a *radial configuration*. This configuration is obtained by opening and closing switches at different points of the mesh network, so that the connected network forms a tree with the power source as its root node; multiple drop-off points are treated as a single power source. Thus, power must flow from the power source to each demand point following a unique path through the power lines, without exceeding line capacities or violating voltage-drop standards.

Hundreds of thousands of industrial, commercial and residential customers may rely on a distribution network to deliver power without interruption. A fault in a major power line of such a network can disconnect thousands of customers and can take hours or even days to repair. In the case of a radial configuration, the failure of a cable will disconnect all customers in the subtree served by this cable. This can cause significant disruption to customers' day-to-day operations and irretrievable financial losses, as happened in Auckland, New Zealand in 1998 (CNN 1998).

In the event of a fault in a radial configuration, the distribution company will commonly reroute flow to restore supply to customers as quickly as possible. (It may be impossible to quickly identify and repair a fault, so rerouting is often the immediate response; repair occurs later.) This rerouting is effected by opening electrical switches so as to isolate the faulted section, and then by closing switches to establish an alternative path for power to flow from the source to affected customers. The rerouting amounts to switching the radial configuration from one tree topology into another. To enable this switching, the company builds redundancy into the distribution network, in the form of lines that are not used under normal circumstances, but are on hand to be used for "recourse", i.e., for recovering a working, radial configuration. The full set of lines forms the "underlying mesh structure."

We say that a (mesh) distribution network is $n - 1$ *survivable* if it has enough

capacity to reroute supply to customers in the case of a fault on any single line. It is clear that any network with nodes of degree 1 will not be $n - 1$ survivable, but the demands of many of these nodes are sufficiently small or remote that we can ignore them for survivability purposes. These nodes and the “spur” lines that connect them to the network can then be absorbed recursively into their upstream nodes, until all nodes have degree at least two in the underlying network. In this process the demand at the downstream nodes on these collapsed spur lines is aggregated and added to the demand at the upstream node. We shall henceforth assume that the mesh networks we deal with have no spur lines.

Industrial customers are willing to pay to ensure that the distribution network they are connected to is $n - 1$ survivable, so we must ensure that it is and remains so in the face of increasing demand. Our time frame is about one year, so the problem we seek to solve is: given peak-demand estimates for one year hence, where should we add capacity to ensure that the distribution network remains $n - 1$ survivable? (We investigate longer-term capacity-planning models, with uncertain demands, in a separate paper, Singh et al. 2004.)

A network design that is $n - 1$ survivable must be protected from a fault in any one of a large number of different lines. For large networks this represents a formidable number of random outcomes to plan for. In fact, given a radial configuration, the protection need only be from single faults in lines that form part of the tree, and in practice the impact of failure will be greatest for lines that carry the largest amounts of power. In radial networks these are the *trunk lines* (or simply “trunks”) that connect each drop-off point into the network. Flow cannot increase as it moves away from the source, so a fault on a downstream line will interrupt no more power flow than a fault on the line’s unique, upstream trunk.

In general we expect the impact of failure to be greatest for lines that carry the largest amounts of power, and so a good starting point for designing a network with $n - 1$ survivability is to first protect the network against faults in the trunk lines. (The

example that we solve in the sequel has 20 such lines.) In fact providing this level of robustness often provides protection against failure in many other lines as a byproduct, so a solution for trunk lines might turn out to be $n - 1$ survivable, or close enough to allow a full solution to be constructed with minimal extra effort. We return to discuss this in the final section of the paper.

Installing capacity in the network requires substantial capital investments, and gains from optimizing investments can be significant. We can increase the capacity of the network by:

1. installing cables along new routes; and
2. replacing an older cable on an existing route by a new higher capacity cable (reinforcement).

Installation of new cables, and even some reinforcements, can also require the installation of ancillary equipment such as transformers and switches. We simply incorporate the cost of such equipment into the associated cable's cost. Installation of small-scale power generators at or near demand points represents an alternative form of capacity expansion which may become relevant in the future. We can model such a generator as a dummy trunk that potentially connects the power source to the generator's connection point in the network, with a cost equal to that of installing the new generator.

This paper develops a model and column-generation solution procedure to determine capacity investments for an electricity distribution network so as to make it $n - 1$ survivable at least capital cost. This problem, which we denote SNDR (survivable network design, radial configuration), is essentially a two-stage stochastic mixed-integer program. The first stage chooses capacity-expansion decisions for each line, or potential line, in the distribution network. After this decision is made, a random outage is observed, and the network is configured into a tree by closing and opening switches so as

to meet demand, if this is possible. In its simplest form there are no probability distributions for this model—we must meet all customer demand in all failure scenarios—but a version of SNDR with costs on unsatisfied demand can be formulated that accounts for failure probabilities.

Because of the complexity imposed by discrete capacity expansions and by the discreteness of radial-configuration requirements, SNDR must incorporate integer variables in both the first and second stages, along with continuous variables in the second stage. Stochastic mixed-integer programs like this are notoriously difficult to solve (Schultz et al. 1995). Our column-generation approach represents a significant advance on the state of the art for solving such problems. We present results that show our methods gives solutions in a reasonable amount of time for real-world problem instances that general-purpose commercial solvers cannot solve.

Survivable network design has been much studied in the literature, predominately for telecommunications networks (e.g., Rajan and Atamtürk 2004; Jothi and Raghavachari 2004; Luss and Wong 2004; Rajan and Atamtürk 2002; Wessaly 2000; Myung, Kim and Tcha 1999; Balakrishnan, Magnanti and Mirchandani 1998; Dahl and Stoer 1998; Newport and Varshney 1991; and Gavish et. al 1989.). Much of this research concerns the dimensioning of ring architectures for transmission networks that connect exchanges or add/drop multiplexers (ADMs), where link failure causes a recourse switching of point-to-point traffic around the opposite side of the ring. However, other research (e.g., Monma and Shallcross 1989, Jothi and Raghavachari 2004) optimizes the design of more general topologies with survivability implied through connectivity and/or capacity constraints.

Column generation has been used for survivable network design in only a few instances and again, the focus has been telecommunications networks. Wessaly (2000) uses column-generation to solve a secondary path-flow model that creates cuts for a primary branch-and-cut algorithm. Dahl and Stoer (1998) take a similar approach, but their primary algorithm is a cutting-plane algorithm. Rajan and Atamtürk (2002) model

the capacity expansion of a network that routes multiple commodities, each representing communication between a unique origin-destination pair. They use column generation to generate the primary paths for the commodities as well as cycles for rerouting flow when edges fail. Rajan and Atamtürk (2004) solve the same problem with a formulation that only requires column generation for cycles.

The use of column generation for solving stochastic integer programs is relatively new: Lulli and Sen (2004) use branch and price (column generation plus branch and bound) for stochastic batch-sizing problems; Shiina and Birge (2004) use column generation to solve a unit-commitment problem under demand uncertainty; Damodaran and Wilhelm (2004) model high-technology product upgrades under uncertain demand and use branch and price as a solution technique; and Silva and Wood (2004) present an efficient branch-and-price approach for a class of two-stage stochastic mixed-integer programs. Our master problem and subproblems are significantly different from those used in any of these papers.

Our model, SNDR, is different from most other survivable network design models. It considers only a single commodity, electrical power, but must also incorporate model constructs to select an operating, radial configuration from the mesh network that it designs, i.e., upgrades and/or expands. Survivability is handled by explicitly modeling potential faults in a set of scenarios, rather than building “slack” into the system or enforcing connectivity-type constraints. As in other models, SNDR can incorporate multiple “technologies” for capacity expansion. From a modeling perspective, these just represent different line capacities that might be installed between two network nodes, each with a different cost. In reality, these can represent different cable sizes, the option to replace an overhead line with an underground line, installation of a new cable plus a transformer, etc.

For simplicity, SNDR ignores one practical consideration that is important for some electricity distribution networks. In particular, it ignores voltage drops. We are currently concerned with urban networks consisting primarily of underground cables

where voltage drops are, in fact, negligible. SNDR will require refinement for other situations. We also assume that each edge will be expanded at most once in our planning horizon using a single technology (any mix of technologies can be modeled by an appropriate labeling of a binary variable).

We note that Nara et al. (1994), Ramirez and Bernal (2001) and Ferreira et al. (2001) do present models similar to SNDR, and they take voltage drops into account. However, these authors offer only heuristics for their models' solutions.

The layout of the paper is as follows. The next section formulates SNDR mathematically. Difficulty in solving this model motivates the column-oriented decomposition and column-generation solution procedure described in section 3. The subproblem in this procedure is a difficult-to-solve mixed-integer program, however, and section 4 shows how to ameliorate this difficulty with a condensed "super-network formulation." Section 5 presents computational results and section 6 presents conclusions.

2 Formulation of SNDR

In an operating, radial configuration of a distribution network, power must flow from the power source along unique paths to the demand points through power lines, without exceeding those lines' capacities. Typically, each power line has two switches, one at either end, which can be closed or opened to allow or disallow power flow, respectively. We refer to a power line with closed switches as *active*, and one with open switches as *inactive*. A distribution network is operated in a radial (tree) configuration by opening and closing switches; only active power lines form the operating configuration.

We model the underlying mesh structure of the network as a connected, undirected graph $G = (\mathcal{N}, \mathcal{E})$ consisting of a set of *nodes* $i \in \mathcal{N}$ and a set of *edges* $e \in \mathcal{E}$ such that $e = (i, j)$, where $i, j \in \mathcal{N}$ and $i \neq j$. A node represents a demand point and/or a junction; an edge represents a power line that connects adjacent nodes.

Power may flow in either direction along a power line, and to model this we

create a directed version of G , denoted $G' = (\mathcal{N}, \mathcal{K})$. The set of nodes in G' is the same as in G , but \mathcal{K} replaces each edge $e = (i, j)$ with two anti-parallel, directed *arcs* (i, j) and (j, i) . For edge $e = (i, j)$, we define $\mathcal{K}_e = \{(i, j), (j, i)\}$, so we may also write $\mathcal{K} = \cup_{e \in \mathcal{E}} \{\mathcal{K}_e\}$.

Actually, if we allowed negative flows, the directed-network model would be unnecessary. However, the directed model also enables constructs in the tree-forming submodel that yield tighter linear-programming relaxations than do its undirected counterparts (Magnanti and Wolsey 1995). Thus, computational efficiency dictates the use of the directed-network model.

We model the power source as node $i_0 \in \mathcal{N}$. Note that, in practice, any flow on the edges incident on i_0 will always be directed away from it. Arcs directed towards i_0 will always have zero flow and can be excluded from the model.

We are concerned with a set of single-fault scenarios $s \in \mathcal{S}$. Each scenario corresponds to the failure of a single edge $e(s)$. For a network to be classified as survivable, we must be able to identify a capacity-feasible radial configuration for $G(s) = (\mathcal{N}, \mathcal{E} \setminus e(s))$ for each $s \in \mathcal{S}$. Note that simulating a fault on an edge is equivalent to forcing it to be inactive.

Figure 1 shows a model of a small distribution network. The solid and dashed lines represent active and inactive edges, respectively. The active edges form the operating radial configuration in which, for example, the flow from node 1 to 3 corresponds to flow on arc $k = (1, 3)$ and edge e_7 . The edges e_1 , e_7 , and e_9 incident on the source node $i_0 = 1$, represent the trunks. A fault on e_7 disconnects supply to node 3; the radial configuration can be restored and flow rerouted to node 3 by activating e_{10} . (The fault on e_7 would be isolated by opening switches not shown, located on that edge near its endpoints.)

We are now ready to present a mathematical formulation of SNDR, which we denote SNDR-0. The reader should note that this model is essentially a two-stage stochastic program with a scenario representation of uncertainty, but with a special

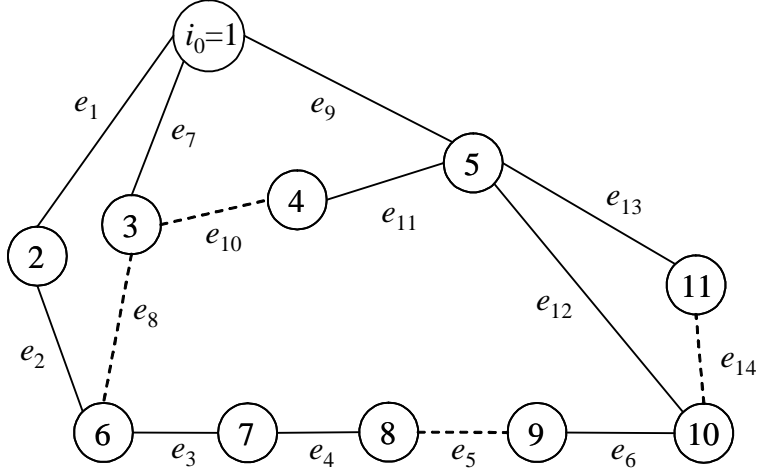


Figure 1: Model of small distribution network.

recourse function: the second-stage cost is 0 if all demand is met, and is infinite if any demand goes unmet.

Sets and Indices

$i \in \mathcal{N}$ nodes in the distribution network.

$e \in \mathcal{E}$ edges in the network

$k \in \mathcal{K}$ anti-parallel arcs corresponding to \mathcal{E}

$k \in \mathcal{K}_e$ pair of anti-parallel arcs representing edge e

$l \in \mathcal{L}_e$ technologies available for capacity expansion of edge e

$s \in \mathcal{S}$ single-edge fault-scenarios

i_0 power source node

Data [units]

A_{ik} 1 if $k = (j, i)$, -1 if $k = (i, j)$, and 0 otherwise

C_{el} cost of expanding capacity on edge e using technology l [\\$]

D_i peak demand at node i [MVA]

$e(s)$	failing edge in scenario s
U_{e0}	initial capacity of edge e [MVA]
U_{el}	additional capacity of edge e if installing technology l [MVA]
U_e	maximum possible capacity for edge e [MVA]

Variables [units]

x_{el}	1 if technology l is chosen for expanding edge e , and 0 otherwise
z_{ks}	1 if arc k is active (part of the operating radial configuration) in scenario s , and 0 otherwise
f_{ks}	power flow on arc k in scenario s [MVA]

Formulation for SNDR-0

$$\min_{x,z,f} \sum_{e \in \mathcal{E}} \sum_{l \in \mathcal{L}} C_{el} x_{el} \quad (1)$$

$$\text{s.t. } f_{ks} \leq U_{e0} + \sum_{l \in \mathcal{L}_e} U_{el} x_{el} \quad \forall e \in \mathcal{E}, k \in \mathcal{K}_e, s \in \mathcal{S}, \quad (2)$$

$$\sum_{l \in \mathcal{L}_e} x_{el} \leq 1 \quad \forall e \in \mathcal{E}, \quad (3)$$

$$\sum_{k \in \mathcal{K}} A_{ik} f_{ks} = D_i \quad \forall i \in \mathcal{N}, s \in \mathcal{S}, \quad (4)$$

$$\sum_{k \in \mathcal{K}: A_{ik}=1} z_{ks} = 1 \quad \forall i \in \mathcal{N} \setminus \{i_0\}, s \in \mathcal{S}, \quad (5)$$

$$\sum_{k \in \mathcal{K}} z_{ks} = |\mathcal{N}| - 1 \quad \forall s \in \mathcal{S}, \quad (6)$$

$$f_{ks} \leq U_e z_{ks} \quad \forall e \in \mathcal{E}, k \in \mathcal{K}_e, s \in \mathcal{S}, \quad (7)$$

$$z_{ks} \equiv 0 \quad \forall s \in \mathcal{S}, k \in \mathcal{K}_{e(s)} \quad (8)$$

$$f_{ks} \geq 0 \quad \forall k \in \mathcal{K}, s \in \mathcal{S}, \quad (9)$$

$$x_{el} \in \{0, 1\} \quad \forall e \in \mathcal{E}, l \in \mathcal{L}_e, \quad (10)$$

$$z_{ks} \in \{0, 1\} \ \forall k \in \mathcal{K}, \ s \in \mathcal{S}. \quad (11)$$

The objective function (1) minimizes the total cost of capacity expansions. Constraints (2) ensure that the flow through any edge does not exceed the edge's total capacity (initial plus added capacity). Note that $U_{e0} = 0$ for new routes that are under consideration by network planners. It would be uneconomical to increase the capacity of an edge more than once during the model's time horizon; constraints (3) impose this restriction. Constraints (4) represent the standard Kirchhoff current-balance (flow-balance) constraints at each node i . Constraints (5) and (6) enforce the radial operating configuration. Constraints (7) ensure that flow is permitted on an arc k if and only if arc k is part of the radial configuration in scenario s , i.e., $z_{ks} = 1$; note that the maximum flow possible on an edge equals the edge's maximum acquirable capacity; thus, with respect to constraints (3), it is sufficient to set the upper bound $U_e = U_{e0} + \max_{l \in \mathcal{L}_e} \{U_{el}\}$. Finally, for each scenario s , constraint (8) simulates a fault on edge $e(s)$ by disallowing flow on arcs $k \in \mathcal{K}_{e(s)}$.

Unfortunately, for real-world problems (e.g., 152 nodes, 182 edges, 5 fault scenarios), this formulation results in a large mixed-integer program (MIP) with a poor LP relaxation, and this MIP is intractable for at least one advanced solver, CPLEX 9.0. The solution difficulties arise, no doubt, from the variable upper-bound constraints (7) as well as the tree-configuration constraints.

Some simple adjustments to this model can modestly tighten its LP relaxation, but experience shows that these changes are insufficient to yield a solvable model. We require the more substantial improvements that accrue from a completely different formulation of SNDR, a column-oriented one. This is the topic of the next section.

3 A Column-Oriented Decomposition

Only the capacity-expansion constraints (2) in SNDR-0 link the fault scenarios. Because of this, SNDR-0 has block-angular structure which motivates the use of a decomposition approach. We use Dantzig-Wolfe decomposition as extended to integer variables (Appelgren 1969). This results in a multi-scenario, column-oriented master problem (MP). For each scenario s , the master problem contains a collection of columns, each of which represents a set of capacity expansions that are sufficient to operate the distribution network for that scenario. We refer to capacity expansions defined by a column as *operating expansions*, and let the set \mathcal{J}_s represent all possible operating-expansion columns for scenario s . The master problem also contains the capacity-expansion variables x_{el} and new variables w_s^j that represent the j^{th} operating-expansion column for scenario s . For each scenario, a constraint forces the selection of exactly one column from the set of possible operating expansions, and another set of constraints allows operating-expansion columns to be selected only if the corresponding capacity expansions have been made. Next, we present the formulation of the column-oriented master problem.

3.1 The Column-Oriented Master Problem

Sets and Indices

$j \in \mathcal{J}_s$ operating-expansion columns for scenario s

$l' \in \mathcal{L}_{el}$ technologies applicable to edge e with capacity
at least as great as technology l

Data

A_{els}^j 1 if operating expansion j expands edge e using technology l ,
under scenario s , and 0 otherwise

Variables

x_{el} 1 if edge e is expanded using technology l , and 0 otherwise

w_s^j 1 if operating-expansion column j is selected for scenario s , and 0 otherwise

Master-Problem Formulation (MP)

$$\min_{x,w} \sum_{e \in \mathcal{E}} \sum_{l \in \mathcal{L}_e} C_{el} x_{el} \quad [\text{dual variables}] \quad (12)$$

$$\text{s.t.} \quad \sum_{l' \in \mathcal{L}_{el}} x_{el'} - \sum_{j \in \mathcal{J}_s} A_{els}^j w_s^j \geq 0 \quad \forall e \in \mathcal{E}, l \in \mathcal{L}_e, s \in \mathcal{S}, \quad [\pi_{els}] \quad (13)$$

$$\sum_{j \in \mathcal{J}_s} w_s^j = 1 \quad \forall s \in \mathcal{S}, \quad [\mu_s] \quad (14)$$

$$w_s^j \in \{0, 1\} \quad \forall s \in \mathcal{S}, j \in \mathcal{J}_s \quad (15)$$

$$x_{el} \in \{0, 1\} \quad \forall e \in \mathcal{E}, l \in \mathcal{L}_e. \quad (16)$$

The MP's objective function (12) minimizes the capacity-expansion costs just as the original model's objective function does. The convexity constraints (14) select exactly one column from the set of possible operating expansions for each scenario s . Constraints (13) ensure that an operating-expansion column is not chosen for any scenario unless sufficient capacity has been installed.

It is impractical to enumerate all possible operating-expansion columns in MP, so we employ *dynamic column generation*: we generate columns “on the fly” through optimization subproblems. To do this, we first create a restricted master problem (RMP) containing a modest-sized subset of all possible columns. Let \mathcal{J}_s represent the current subset of operating-expansion columns for scenario s .

The column-generation technique solves the LP relaxation of the RMP and extracts the corresponding optimal dual variables $\hat{\pi}_{els}$ and $\hat{\mu}_s$ from the master problem. The *column-generation subproblem* then uses those values in an attempt to construct one or more columns with negative reduced cost for the MP; separate subproblems can be constructed for each scenario. If a favorable column is found, it is inserted into the the RMP, which is then re-solved. The cycle of solving subproblems and master problems repeats until no favorable column can be identified. At that point, we know that we have solved the LP relaxation of the MP, and if that solution happens to be

integer, we have solved SNDR. (We refer the reader to Barnhart et al. (1998) for a comprehensive discussion of column generation, and to Lübbecke and Desrosiers (2004) for a compendium of column-generation applications.)

The subproblem for each scenario s contains the operational and demand constraints for the distribution network, as well as the fault-simulation constraint (8) specific to scenario s that forces the failed edge $e(s)$ to be inactive, i.e., $z_{ks} \equiv 0$, $k \in \mathcal{K}_{e(s)}$. In essence then, the subproblems represent single-scenario capacity-expansion problems that find the minimum-cost capacity expansions for a distribution network that does not contain the the failed edge $e(s)$. (Of course, the costs are modified by the current dual variables.) Note that these capacity expansions are made with respect to all distribution-network operating constraints, and hence our use of the term “operating expansions” in the RMP.

3.2 Column-Generation Subproblem

We use the edge capacity expansions given by the subproblem (SP) solution to construct the operating-expansion columns of the RMP. The SP formulation is the same as a single-fault scenario SNDR-0 formulation, except that the objective of SP incorporates dual-variable values $\hat{\pi}_{els}$ and $\hat{\mu}_s$ from the master problem. The subproblem is:

Subproblem Formulation (SP(s))

$$\min_{x,z,f} \sum_{e \in \mathcal{E}} \sum_{l \in \mathcal{L}_e} \hat{\pi}_{els} x_{el} - \hat{\mu}_s \quad (17)$$

s.t. (2) – (11) for fixed scenario s

We have successfully solved small problems using the column-generation technique outlined. And, we invariably obtain integer solutions for the optimized RMP, so there seems to be no need for a complete “branch-and-price algorithm” to solve these problems. (Branch and price embeds column generation within a branch-and-bound

algorithm; see Barnhart et al. 1998. We discuss this general topic in more detail in section 6.) For larger, real-world problems, the column-generating subproblems solve in a reasonable amount of time in early iterations of the procedure, but these times become prohibitive in later iterations. Other researchers observe a similar slow-down in subproblem solution times as the dual variables converge to their optimal values (Vanderbeck and Wolsey 1996). We overcome this difficulty with a stronger formulation of the SNDR-0 subproblem, as described in the next section.

4 A Super-Network Formulation

In some integer-programming problems, a careful choice of the information that a variable represents can significantly tighten the LP relaxation of the the problem (Nemhauser and Wolsey 1988, pp. 14-17). It is logical in SNDR-0 to have variables that correspond to edges, but we will see here that a more compact representation of the network and associated decisions variables leads to a tighter formulation. In particular, we will exploit the sparse nature of the distribution network’s underlying mesh structure along with the requirement that the network operate as a tree.

4.1 The Super-Network

Many nodes in a distribution network will have degree of 2; we call these *sub-nodes*, and refer to all nodes with degree 3 or greater as *super-nodes*. (All nodes with degree 1 have been recursively collapsed into a sub-node or super-node.) Let $\mathcal{M} \subseteq \mathcal{N}$ denote the set of all super-nodes. We say that two super-nodes i and j are *adjacent* if they are joined by a chain in which all nodes except i and j are sub-nodes. We denote this set of sub-nodes by \mathcal{N}_{ij} and let \mathcal{E}_{ij} denote the edges in the chain joining i and j . In the *super-network*, any chain joining two super-nodes i and j is represented by two anti-parallel *super-arcs* $k = (i, j)$ and $k' = (j, i)$. We say that the nodes in \mathcal{N}_{ij} and edges in \mathcal{E}_{ij} are *spanned* by

the super-arc k (or k').

To illustrate, consider Figure 2(a) which extracts a small portion of the network in Figure 1 (in which $\mathcal{M} = \{1, 3, 5, 6, 10\}$). That portion of the network contains super-nodes 6 and 10 for which we define $\mathcal{E}_{6,10} = \{e_3, e_4, e_5, e_6\}$ and $\mathcal{N}_{6,10} = \{7, 8, 9\}$. Figure 2(b) shows the super-arcs $k = (6, 10)$ and $k' = (10, 6)$ that span $\mathcal{N}_{6,10}$ and $\mathcal{E}_{6,10}$.

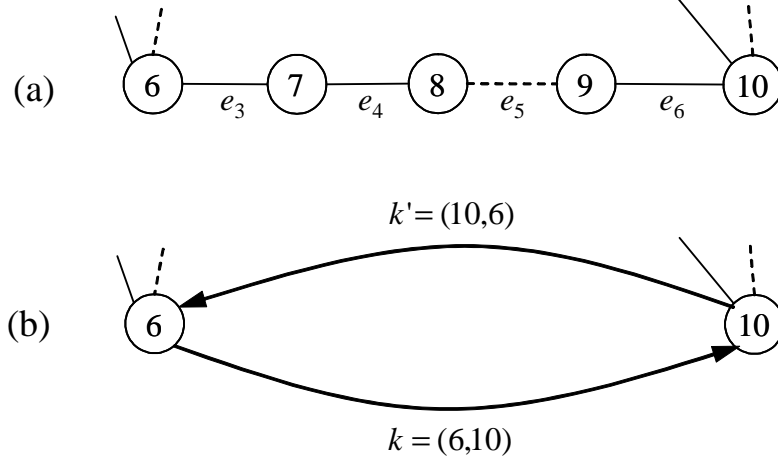


Figure 2: (a) Super-node pair 6 and 10 associated with edges $\mathcal{E}_{6,10} = \{e_3, e_4, e_5, e_6\}$ and sub-nodes $\mathcal{N}_{6,10} = \{8, 9, 10\}$. (b) The directed super-arcs k and k' span edges $\mathcal{E}_{6,10}$ and sub-nodes $\mathcal{N}_{6,10}$ in the super-network.

In SNDR-0, for a given scenario s , each edge e is represented by two flow variables, f_{ks} , $k \in \mathcal{K}_e$, two “tree variables” z_{ks} , $k \in \mathcal{K}_e$, and one capacity-expansion variable x_{el} for each $l \in \mathcal{L}_e$. Thus, if $|\mathcal{L}_e| = 1$ for all $e \in \mathcal{E}_{6,10}$ in Figure 2(a), 20 variables in SNDR-0 would result. In the super-network model below, SNDR-SN, there will be one flow variable and one tree variable for each super-arc, one capacity-expansion variable for each spanned edge and one “break-edge variable,” described below, for each spanned edge. Thus, the portion of the super-network shown in Figure 2(b) will only account for 12 variables.

To develop this model and underlying concepts further, we shall restrict attention to the nontrivial case where $|\mathcal{E}_{ij}| > 1$. Given a pair of adjacent super-nodes i and j , and a feasible radial configuration, we know that either:

1. all edges $e \in \mathcal{E}_{ij}$ are active, or
2. exactly one edge $e' \in \mathcal{E}_{ij}$ is inactive.

For case 1), we know that power will flow through all the edges in either one of two directions, and we can model this as flow on super-arcs. The flow on either one of these super-arcs represents a flow on the corresponding edges $e \in \mathcal{E}_{ij}$ of the super-network. By an abuse of notation, we refer to a super-arc with nonzero flow as active, and its anti-parallel partner as inactive.

For case 2), the inactive edge e' “breaks the super-arc” in the sense that no flow through either super-arc (i, j) or (j, i) can occur. Now both super-arcs are said to be inactive. We refer to the inactive edge as a *break-edge*. The dashed edge e_5 in Figure 2(a) represents such an edge.

In addition to reducing the number of variables compared to SNDR-0, we will see that the super-network representation eliminates the need for flow-balance constraints at the sub-nodes, resulting in a much smaller model. Furthermore, opportunities for tightening the super-network model are easier to identify and implement.

4.2 SNDR-SN: A Super-Network Formulation of SNDR

Next, we present SNDR-SN, the super-network formulation of SNDR.

Sets and Indices

$$\begin{aligned}
i &\in \mathcal{N} && \text{nodes} \\
m &\in \mathcal{M} \subseteq \mathcal{N} && \text{super-nodes (nodes with degree } \geq 3) \\
k &\in \mathcal{A} && \text{super-arcs (spanning super-nodes)} \\
i &\in \mathcal{N}_k \subseteq \mathcal{N} && \text{sub-nodes spanned by super-arc } k
\end{aligned}$$

$k \in \mathcal{RS}_m$	all super-arcs entering super-node m (reverse star)
$k \in \mathcal{FS}_m$	all super-arcs leaving super-node m (forward star)
$e \in \mathcal{E}_k^1$	edges spanned by super-arc k ($\cup_{k \in \mathcal{A}} \mathcal{E}_k^1 = \mathcal{E}$)
$k \in \mathcal{A}^1$	super-arc k , with $(k+1)^{st}$ super-arc in anti-parallel
$k \in \mathcal{A}^2$	super-arcs k , which when broken, result in flow that forces an expansion on an edge $e \in \mathcal{E}_k^1$
$e' \in \mathcal{E}_{ke}^2$	edges $e' \in \mathcal{E}_k^1$, which when broken, result in flow that forces an expansion on the edge $e \in \mathcal{E}_k^1$
i_0	source node (always a super-node)
$\underline{m}(k)$	tail super-node of super-arc k
$\overline{m}(k)$	head super-node of super-arc k
$i(e, k)$	end node of edge e closest to $\underline{m}(k)$, e.g., for $k = (6, 10)$ and $e = 4$ in Figure 2, $i(e, k) = 7$.

Data [units]

C_{el}	cost of expanding capacity on edge e using technology l [\$]
D_i	peak demand at node i [MVA]
D_k^1	total peak demand for all sub-nodes between super-nodes $\underline{m}(k)$ and $\overline{m}(k)$ [MVA], e.g., for $k = (6, 10)$, $D_k^1 = \sum_{i \in \mathcal{N}_{(6,10)}} D_i$ (see Figures 2 and 3)
D_{ek}^2	total peak demand for sub-nodes of arc k between $\underline{m}(k)$ and up to and including sub-node $i(e, k)$ [MVA], e.g., for $k = (6, 10)$ and $e = 5$, $i(e, k) = 8 \Rightarrow D_{ek}^2 = D_7 + D_8$ (see Figures 2 and 3)

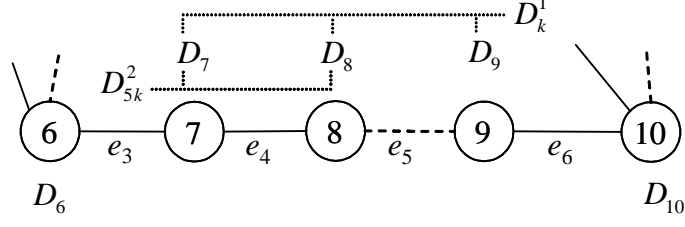


Figure 3: Section of a network to illustrate some of the notation used in the super-network formulation

U_{e0} initial capacity of edge e [MVA]

U_{el} additional capacity on edge e if installing technology l [MVA]

Variables [units]

x_{el} 1 if edge e is expanded using technology l , and 0 otherwise

b_{es} 1 if edge e is inactive in scenario s , and 0 otherwise (break-edge variable)

z_{ks} 1 if super-arc k is active in scenario s , and 0 otherwise

f_{ks} flow on super-arc k in scenario s [MVA]

Formulation (SNDR-SN)

(A one-line explanation of each constraint has been included to give the reader the general idea of the constraint without having to refer to the detailed description that follows the formulation.)

$$\min_{x,b,z,f} \sum_{e \in \mathcal{E}} \sum_{l \in \mathcal{L}_e} C_{el} x_{el} \quad (18)$$

s.t Maximum of one expansion for each edge:

$$\sum_{l \in \mathcal{L}_e} x_{el} \leq 1 \quad \forall e \in \mathcal{E}, \quad (19)$$

Super-arc flow capacity-expansion constraints:

$$f_{ks} - D_{ek}^2 z_{ks} \leq U_{e0} z_{ks} + \sum_{l \in \mathcal{L}_e} U_{el} x_{el} \quad \forall s \in \mathcal{S}, k \in \mathcal{A}, e \in \mathcal{E}_k^1, \quad (20)$$

Flow-balance constraints:

$$\sum_{k \in \mathcal{RS}_m} (f_{ks} - D_k^1 z_{ks}) - \sum_{k \in \mathcal{FS}_m} f_{ks} - \sum_{k \in \mathcal{FS}_m} \sum_{e \in \mathcal{E}_k^1} D_{ek}^2 b_{es} = D_m$$

$$\forall s \in \mathcal{S}, m \in \mathcal{M} \setminus \{i_0\}, \quad (21)$$

Exactly one edge spanned by a super-arc is broken or all edges are active:

$$z_{ks} + z_{k+1,s} + \sum_{e \in \mathcal{E}_k^1} b_{es} = 1 \quad \forall s \in \mathcal{S}, k \in \mathcal{A}^1, \quad (22)$$

Flow in tree (feasible configuration):

$$f_{ks} \leq U_k z_{ks} \quad \forall s \in \mathcal{S}, k \in \mathcal{A}, \quad (23)$$

$$\text{where } U_k = \min_{e \in \mathcal{E}_k^1} \left\{ D_{ek}^2 + U_{e0} + \max_{l \in \mathcal{L}_e} U_{el} \right\}.$$

Tree constraint 1:

$$\sum_{k \in \mathcal{A}} z_{ks} = |\mathcal{M}| - 1 \quad \forall s \in \mathcal{S}, \quad (24)$$

Tree constraint 2:

$$\sum_{k \in \mathcal{RS}_m} z_{ks} = 1, \quad s \in \mathcal{S} \quad \forall m \in \mathcal{M} \setminus \{i_0\}, \quad (25)$$

Expansions due to breaks in super-arcs:

$$\sum_{e' \in \mathcal{E}_{ke}^2} b_{e's} \leq \sum_{l \in \mathcal{L}_e} x_{el} \quad \forall s \in \mathcal{S}, k \in \mathcal{A}^2, e \in \mathcal{E}_k^1, \quad (26)$$

Fault-simulation constraints:

$$b_{e(s)s} \equiv 1 \quad \forall s \in \mathcal{S}, \quad (27)$$

Domain restrictions on variables:

$$f_{ks} \geq 0 \quad \forall k \in \mathcal{A}, s \in \mathcal{S}, \quad (28)$$

$$b_{es} \in \{0, 1\} \quad \forall e \in \mathcal{E}, \quad s \in \mathcal{S}, \quad (29)$$

$$x_{el} \in \{0, 1\} \quad \forall e \in \mathcal{E}, \quad l \in \mathcal{L}_e, \quad (30)$$

$$z_{ks} \in \{0, 1\} \quad \forall k \in \mathcal{A}, \quad s \in \mathcal{S}. \quad (31)$$

The objective function (18) minimizes the total cost of capacity expansion. Similar to the previous formulation, constraints (19) allow at most one capacity expansion on any edge.

This formulation does not explicitly model flows on the edges. Instead, we compute them using the super-arc flows f_{ks} . To be more precise, the flow on the first edge that a super-arc spans equals the super-arc flow f_{ks} ; for super-arcs that span more than one edge, the flow on each edge is calculated by subtracting the upstream demand D_{ek}^2 from the super-arc flow f_{ks} ; this is shown on the left-hand-side of the super-arc capacity-expansion constraints (20). This forces expansion on an edge if the edge flow is greater than its initial capacity U_{e0} .

Constraints (22) and (23) ensure that when a break-edge e breaks a super-arc k ($b_{es} = 1$), then the corresponding super-arc flow is zero. Furthermore, (22) ensures that there is at most one active super-arc between any pair of super-nodes. It is important to observe however, that if $|\mathcal{E}_k^1| > 1$ and a break-edge e breaks a super-arc k , then $f_{ks} = 0$, but (implicit) flow on edges $\bar{e} \in \mathcal{E}_k^1 \setminus \{e\}$ is likely to occur. In our model, a preprocessing step adds a break-edge expansion constraint to the model when a break in edge $e \in \mathcal{E}_k^1$ results in a flow on an adjacent edge $\bar{e} \in \mathcal{E}_k^1$ that exceeds edge \bar{e} 's initial capacity $U_{\bar{e}0}$. These constraints force an expansion on edge \bar{e} when there is a break on edge e . In some instances when $|\mathcal{E}_k^1| > 1$, breaks in several different edges $e \in \mathcal{E}_k^1$ may result in the creation of several break-edge expansion constraints for the same adjacent edge $\bar{e} \in \mathcal{E}_k^1$. In such cases, it is possible to aggregate these expansion constraints to derive a stronger constraint, as in constraints (26).

We enforce flow-balance constraints (21) only at super-nodes in SNDR-SN. These

flow-balance constraints have an additional “flow-out” term $(\sum_{k \in \mathcal{FS}_m} \sum_{e \in \mathcal{E}_k^1} D_{ek}^2 b_{es})$ that constitutes the flow needed to satisfy demand of sub-nodes up to the break-edge on each inactive (“broken”) super-arc $k \in \mathcal{FS}_m$.

Constraints (24) and (25) ensure that the super-network satisfies the radial configuration requirement by forcing the set of active super-arcs ($z_{ks} = 1$) to form a “super-tree”. Notice that similar to flow-balance constraints, constraints (25) are also only defined at super-nodes, which results in fewer constraints than in SNDR-0.

For each fault scenario s , constraint (27) simulates a fault on edge $e(s)$ by breaking it, i.e., forcing it to be inactive.

4.3 Strengthening SNDR-SN

As mentioned earlier, we can pre-compute the minimum flow on a super-arc if it is used. This allows us to define additional constraints which may tighten the model’s LP relaxation. For example, if super-arc k is active, then the minimum flow f_{ks} (which is leaving $\underline{m}(k)$) is bounded below by $D_k^1 + D_{\overline{m}(k)}$, i.e., the total demand for all subnodes in \mathcal{N}_k plus the demand at the head node $\overline{m}(k)$ of super-arc k . We use this information to impose lower-bounding constraints such as:

$$f_{ks} \geq (D_k^1 + D_{\overline{m}(k)})z_{ks} \quad \forall s \in \mathcal{S}, k \in \mathcal{A}. \quad (32)$$

In addition, we use this information to compute the minimum required flow through the edge $e \in \mathcal{E}_k^1$ if super-arc k is used, and define capacity-expansion constraints that force expansions on edges e if the flow on them exceeds their initial capacities U_{e0} . Such constraints are defined by:

$$z_{ks} \leq \sum_{l \in \mathcal{L}_e} x_{el} \quad \forall s \in \mathcal{S}, k \in \mathcal{A}^3, e \in \mathcal{E}_k^3, \quad (33)$$

where the set \mathcal{A}^3 represents super-arcs k , which when active, result in flow that force expansion on edges $e \in \mathcal{E}_k^1$, and the set \mathcal{E}_k^3 denotes edges $e \in \mathcal{E}_k^1$ requiring expansion if

super-arc $k \in \mathcal{A}^3$ is active.

Additional improvements in the LP relaxation are made by multiplying the coefficients D_{ek}^2 and U_{e0} by z_{ks} in the capacity-expansion constraints (20). (Constraints (2) in SNDR-0 can also be strengthened by multiplying U_{e0} with z_{ks} , but this yields only minor improvements in solution times.)

We will demonstrate the advantages of the super-network constructs, and the strengthening just described, in the following section.

5 Computational Experiments

This section demonstrates the relative computational performance of the models and solution procedures described in this paper. We use the following abbreviations:

SNDR-0	the original, extensive model
SNDR-SN	the super-network extensive model
SNDR-SN _S	SNDR-SN with strengthening as described in section 4.3
CG-0	column generation using subproblems derived from SNDR-0
CG-SN _S	column generation using subproblems derived from SNDR-SN _S

All problem instances derive from data for a distribution network in New Zealand. The actual network supplies power to an urban area that contains mostly large industrial and commercial customers who pay extra fees for a high level of reliability, i.e., for an $n - 1$ survivable network. Planning is done such that the network remains survivable for at least one year into the future. Thus, we use peak-demand data that is forecasted one year forward. This forecast may include entirely new demand, e.g., a new residential subdivision, and the data for such situations also describes possible routes for new cables to serve that demand. These are instances of network edges with zero initial capacity.

The network data comprise 152 nodes, most of which are demand points, and

182 edges. (The true network contains more nodes and edges, but spur lines have been collapsed in the data.) Four demand points represent completely new demand, and 14 edges represent completely new cable routes. The power source has 20 trunk lines. Three of the trunks correspond to new cable routes. We model a single capacity-expansion technology for each edge and consider faults only on trunks. The super-network representation of this problem has only 32 super-nodes and 124 super-arcs. For testing, a set of problem instances is obtained by varying the number of fault-scenarios, each of which corresponds to the failure of a single trunk line.

We solve (attempt to solve) all problems using a desktop computer with a Pentium 4, a 2.6 GHz processor, and 1 GB of RAM. We generate all models, and implement our decomposition algorithms within the Mosel algebraic modeling system, version 1.24, from Dash Optimization. The LP master problems are solved with the Xpress-MP version 14.24 LP solver, also from Dash Optimization, but the MIP subproblems and extensive models are solved with CPLEX, version 9.0 from ILOG, Inc. All problems are solved with a relative optimality tolerance of 0.05% and each run is limited to 7,200 seconds. Our column-generation algorithm incorporates the duals-stabilization procedure described by du Merle et. al (1999). Solver settings are constant throughout. All MIPs are solved with default parameter settings except that Gomory cuts are turned off and a moderate level of probing is used ($\text{CPX_PARAM_PROBE} = 2$).

Table 1 displays computational results for 12 different problem instances. We attempt to solve each instance with the five solution approaches outlined above. The results summarize easily. The super-network model for SNDR, SNDR-SN, is faster than the original model SNDR-0, and the strengthened super-network model SNDR-SN_S is faster yet. Column generation with strengthened super-network subproblems (CG-SN_S), is vastly more efficient than the other solution methods, and results listed under “CG-0” provide clear evidence that the the strengthened super-network constructs are critical to this efficiency.

Fault Scenarios (number)	SNDR-0 (CPU sec.)	SNDR-SN (CPU sec.)	SNDR-SN _S (CPU sec.)	CG-0 (CPU sec.)	CG-SN _S (CPU sec.)
1	5.1	4.5	2.1	11.3	4.1
2	102.8	7.6	7.5	116.9	35.4
3	458.8	135.1	21.9	224.6	92.9
4	-	2249.3	211.4	1122.0	381.5
5	-	1789.9	2039.0	-	617.4
6	-	-	762.3	-	441.1
7	-	-	3331.5	-	683.4
8	-	-	5285.1	-	2378.5
9	-	-	-	-	2240.4
10	-	-	-	-	4612.4
15	-	-	-	-	1542.4
20	-	-	-	-	1174.0

Table 1: Solution times for SNDR-0, SNDR with an unstrengthened super-network formulation (SNDR-SN), with a strengthened super-network formulation (SNDR-SN_S), column generation with subproblems based on SNDR-0 (CG-0), and column generation with subproblems based on SNDR-SN_S (CG-SN_S). A dash indicates the problem cannot be solved in under 7,200 seconds.

We also note the strength of the LP relaxations of SNDR-0, SNDR-SN and SNDR-SN_S. For the first three problems, which all three can solve, the optimal LP objective value for SNDR-SN improves 4.5% over SNDR-0's, and that improvement is 98.2% for SNDR-SN_S. Both these numbers and the results in the table show that the super-network formulation contributes to efficient solutions of SNDR, but that the additional strengthening is critical for success.

6 Integer Solutions and Fractional Solutions

For the full-scale problem instances we have solved, final LP solutions are invariably integral. Consequently, we have not required a full branch-and-price solution procedure. It is interesting to note, however, that fractional solutions are possible; Figure 4 displays one such instance.

We believe that the master problem is NP-complete, in general, and that the optimal integral solutions we observe must result from an interplay between structure

	x_1	x_2	x_3	x_4	x_5	x_6	w_1^1	w_1^2	w_2^1	w_2^2		
e_1	1						-1	0			\geq	0
e_2		1					0	-1			\geq	0
e_3			1				-1	0			\geq	0
e_4				1			0	-1			\geq	0
e_5					1		0	0			\geq	0
e_6						1	0	0			\geq	0
e_1	1								0	-1	\geq	0
e_2		1							-1	0	\geq	0
e_3			1						-1	-1	\geq	0
e_4				1					0	-1	\geq	0
e_5					1				0	0	\geq	0
e_6						1			0	0	\geq	0
$s=1$							1	1			$=$	1
$s=2$									1	1	$=$	1

Figure 4: An RMP with a fractional optimal solution. With unit cost for each capacity expansion, the optimal solution sets all variables to $1/2$.

and costs. We are still investigating these issues. If the need should arise, we are prepared to implement branch and price using code based on the COIN-OR libraries (Ralphs and Ladanyi 2001) as developed by Silva and Wood (2004).

7 Conclusions

We have described a model, SNDR, along with several formulations, for the design of survivable electricity distribution networks. The model may be viewed as a two-stage stochastic program with a special recourse function. We have also developed a column-generation procedure for solving one of the formulations efficiently. The effectiveness of this solution procedure relies heavily on modeling improvements that strengthen the formulation of the column-generation subproblems. These improvements involve modeling the network structure through a condensed construct, a “super-network,” which leads to smaller subproblems with tighter linear-programming relaxations. This super-network lends itself to further strengthening.

Opportunities exist to increase the speed of this algorithm even further. For instance, variable upper-bound constraints (e.g., constraints (7)), controlled by binary variables tend to produce fractional solutions in the subproblems if upper bounds on arc power flows are not tight. These bounds can be improved by approximately solving deterministic variants of SNDR that maximize arc flow. Preliminary experiments show that this preprocessing does, indeed, improve the algorithm’s performance, and at an acceptable computational cost.

In our computational experiments, we have restricted attention to faults on trunk lines. We have no guarantee that protection against any trunk fault will provide $n - 1$ survivability. However, it is interesting to observe the decrease in computational effort as the number of fault scenarios increases. In these cases, many of the solutions to scenario subproblems return null columns after early column-generation iterations, indicating that the current solution is feasible in this scenario. As more fault scenarios are added we expect this behavior to become more common. This raises the possibility of solving SNDR starting with a few scenarios and then adding violated scenarios “on the fly.” Preliminary experiments with this approach have shown promise.

SNDR represents single failures on lines using scenarios. Recourse decisions reconfigure the underlying mesh network, without the faulted line, into an alternative radial (tree) topology. The cost of this recourse is 0 if a feasible configuration can be found, and is infinity otherwise. A possible extension of SNDR would admit the possibility of shedding customer load as a recourse decision in the event of a line failure. If we (a) let v_{is} represent the amount of load shed at node i under scenario s , (b) let q_i denote the unit penalty for shedding at i , (c) modify demand (flow-balance) constraints to admit penalized load-shedding, and (d) let p_s denote the probability of scenario s , then we can create a standard two-stage stochastic program having objective function

$$\min \sum_{e \in \mathcal{E}} \sum_{l \in \mathcal{L}} C_{el} x_{el} + \sum_{s \in \mathcal{S}} p_s \sum_{i \in \mathcal{N}} q_i v_{is},$$

where the first term represents first-stage capacity costs and decisions, just as in SNDR.

This objective minimizes capacity-expansion costs plus expected penalties. Our column-generation solution procedure, i.e., Dantzig-Wolfe decomposition, follows through for this model, except that the super-network improvements may no longer be valid. In particular, a super-arc being active no longer forces minimum flow quantities through the edges it spans, and thus the flow on each individual edge may need to be tracked.

We conclude by remarking that the column-generation technique for SNDR can be extended to a multi-stage capacity-expansion planning model. In its simplest form this has a scenario-tree representation of uncertainty in demand and no link failures, giving a restricted master problem that is a multi-stage stochastic mixed-integer program. (The multi-stage problem with deterministic demand and link failures has been studied by Kuwabara and Nara 1997 who describe a heuristic solution procedure.) Like the RMP above, many instances of this problem have naturally occurring integer solutions, so it is amenable to solution by standard techniques for stochastic linear programming (see Birge and Louveaux 1997, pp. 155-197). The subproblems are identical to $SP(s)$, and can be strengthened using the “super-network approach”. The details of this implementation and its computational performance are described in Singh, Philpott and Wood (2004).

References

- [1] Ahuja, R.K, Magnanti, T.L and Orlin, J.B *Network Flows: Theory, Algorithms, and Applications*. Prentice-Hall, Englewood Cliffs, New Jersey (1993).
- [2] Appelgren, L. H. “A Column Generation Algorithm For a Ship Scheduling Problem.” *Transportation Science* **3** (1969), 53-68.
- [3] Barnhart, C., Johnson, E. L., Nemhauser, G. L., Savelsbergh, M. W. P. and Vance, P. H. “Branch-and-price: Column Generation for Solving Huge Integer Programs.” *Operations Research* **46** (1998), 316-329.

- [4] Birge, J.R., and Louveaux, F. *Introduction to Stochastic Programming*, Springer-Verlag, New York, 1997.
- [5] CNN. CNN Interactive, “Power outage hits Auckland hours after crisis declared over.” March 27, 1998, <http://www.cnn.com/WORLD/9803/27/auckland.outage>.
- [6] Dahl, G. and Stoer, M. “A Cutting Plane Algorithm for Multicommodity Survivable Network Design Problems.” *INFORMS Journal on Computing* **10** (1998), 1-11.
- [7] Damodaran, P. and Wilhelm, W. “Branch-and-Price Methods for Prescribing Profitable Upgrades of High-Technology Products with Stochastic Demands.” *Decision Sciences* **35** (2004), 55-81.
- [8] Dantzig, G.B. and Wolfe, P. “Decomposition Principle For Linear Programs.” *Operations Research* **8** (1960), 101-111.
- [9] Lakervi, E. and Holmes, E.J. *Electricity Distribution Network Design*, Peter Peregrinus, London, 1995.
- [10] Lübbecke, M.E. and Desrosiers, J. “Selected Topics in Column Generation.” *Les Cahiers de GERAD G-2002-64*, Group for Research in Decision Analysis, Montreal, Canada, 2004.
- [11] du Merle, O., Villeneuve, D., Desrosiers, J., and Hansen, P. “Stabilized Column Generation.” *Discrete Mathematics*, **194** (1999), 229-237.
- [12] Balakrishnan, A., Magnanti, T. and Mirchandani, P. “Designing Hierarchical Survivable Networks.” *Operations Research* **46** (1998), 116-136.
- [13] Ferriera, L.A.F.M., Carvalho, P.M.S., Grave, S.N.C., Barruncho, L.M.F., Jorge, L.A., Quaresma, E., Martins, J.A., Branco, F.C., and Mira, F. “Optimal Distribution Planning-Increasing Capacity and Improving Efficiency and Reliability With

- Minimal-Cost Robust Investment.” *Electricity Distribution, 2001. Part 1: Contributions*. CIRED. 16th International Conference and Exhibition on (IEE Conf. Publ No. 482), **5** (2001).
- [14] Gavish, B., Trudeau, P., Dror, M., Gendreau, M. and Mason, L. “Fiberoptic Circuit Network Design Under Reliability Constraints.” *IEEE Journal on Selected Areas in Communications* **7** (1989), 1181-1187.
- [15] Jothi, R. and Raghavachari, B. “Survivable Network Design: The Capacitated Minimum Spanning Network Problem.” *Information Processing Letters* **91** (2004), 183-190.
- [16] Kuwabara, H. and Nara, K. “Multi-Year and Multi-State Distribution Systems Expansion Planning by Multi-Stage Branch Exchange.” *IEEE Transactions on Power Delivery* **12** (1997), 457-463.
- [17] Lulli, G. and Sen, S. “A Branch-and-Price Algorithm for Multistage Stochastic Integer Programming with Application to Stochastic Batch-Sizing Problems.” *Management Science* **50** (2004), 786-796.
- [18] Luss, H and Wong, R.T. “Survivable Telecommunications Network Design Under Different Types of Failures.” *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans* **34** (2004), 521-530.
- [19] Magnanti, T.L. and Wolsey, L.A. “Design of Survivable Networks.” in *Handbook in Operations Research and Management Science, Volume 7: Network Models* M.O. Ball, T.L. Magnanti, C.L. Monma and G.L. Nemhauser eds., Elsevier Publishing Company, Amsterdam (1995), 503-616.
- [20] Monma, C.L. and Shallcross, D.F. “Methods for Designing Communication Networks with Certain Two-Connected Survivability Constraints.” *Operations Research* **37** (1989), 531-541.

- [21] Myung, Y., Kim, H. and Tcha, D. “Design of Communication Networks with Survivability Constraints.” *Management Science* **45** (1999), 238-252.
- [22] Nara, K., Kuwabara, H., Kitagawa, M. and Ohtaka, K. “Algorithm for Expansion Planning in Distributions Systems Taking Faults into Consideration.” *IEEE Transactions on Power Systems* **9** (1994), 324-330.
- [23] Nemhauser, G.L. and Wolsey, L.A. *Integer and Combinatorial Optimization* John Wiley & Sons, New York, 1988.
- [24] Newport, K.T and Varshney, P.K. “Design of Survivable Communications Networks under Performance Constraints.” *IEEE Transactions on Reliability* **40** (1991), 433-440.
- [25] Orlowski, S. and Wessaly, R. “Hop Limits in Network Design: What is the Impact on Optimal Network Cost?” in review, (2004).
- [26] Rajan, D. and Atamtürk, A. “Survivable Network Design : Routing of Flows and Slacks.” in *Telecommunications Network Design and Management*, G. Anandalingam and S. Raghavan, eds., 65-81, Kluwer Academic Publishers, New York, 2002.
- [27] Rajan, S. and Atamtürk, A. “A Directed Cycle-Based Column-and-Cut Generation Method for Capacitated Survivable Network Design.” *Networks* **43** (2004), 201-211.
- [28] Ralphs, T. K. and L. Ladanyi. *COIN/BCP User’s Manual*. (2001).
- [29] Ramirez-Rosado, I.J., Bernal-Agustin, J.L., “Reliability and Costs Optimization for Distribution Networks Expansion Using an Evolutionary Algorithm.” *IEEE Transactions on Power Systems* **16** (2001), 111-118.
- [30] Schultz, R., Stougie, L. and van der Vlerk, M.H. “Two-stage Stochastic Integer Programming: A Survey.” *Statistica Neerlandica*, **50** (1996), 404-416.

- [31] Shiina, T. and Birge, J.R. “Stochastic Unit Commitment Problem.” *International Transactions in Operational Research* **11** (2004), 19-32.
- [32] Silva, E.F. and Wood, R.K. “Solving a Class of Stochastic Mixed-Integer Programs by Branch-and-Price.” Working paper, Naval Postgraduate School, Monterey, California (2004).
- [33] Singh, K., Philpott, A. and Wood, K. “Multi-Stage Stochastic Integer Programming for Capacity Expansion Planning: A Column-Generation Approach.” Working paper, Department of Engineering Science, University of Auckland (2004)
- [34] Vanderbeck, F. and Wolsey, L. A. “An Exact Algorithm for IP Column Generation.” *Operations Research Letters* **19** (1996), 151-159.
- [35] Wessaly, R. “Dimensioning Survivable Capacitated Networks.” PhD thesis, Technische Universität Berlin, April 2000.